

Web Dev Day 2: HTML5 Guide

DEV dev.to/_bhupeshk_/web-dev-day-2-html5-guide-57nn



Bhupesh Kumar

Posted on Feb 27, 2024 • Edited on May 16, 2024

[#webdev](#) [#html](#) [#beginners](#) [#programming](#)

What is HTML?

HTML, which stands for Hypertext Markup Language, is the standard markup language used to create and design documents on the World Wide Web. It serves as the backbone for structuring and presenting content on web pages. HTML documents are essentially text files that contain tags, which are surrounded by angle brackets (< >), and these tags provide instructions to web browsers on how to display the content.

HTML Elements:

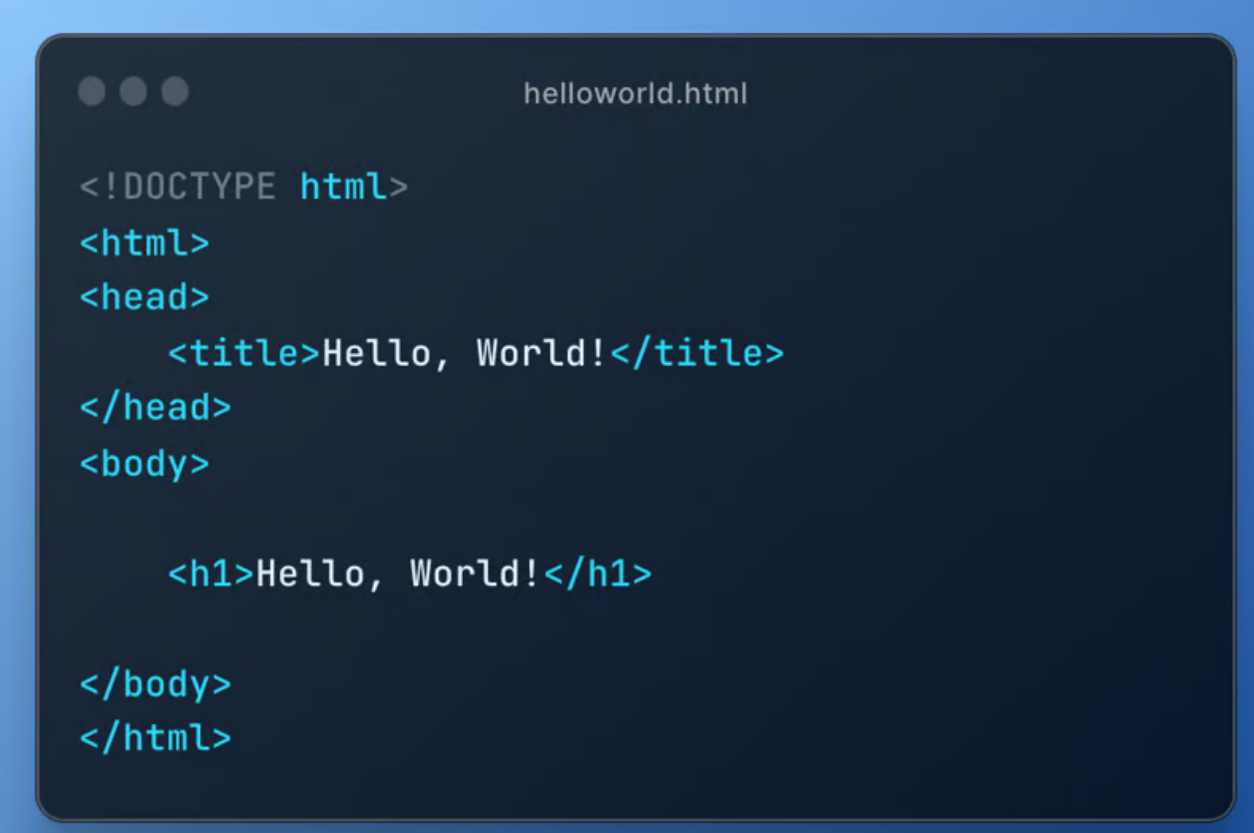
HTML elements are the building blocks of an HTML document. Each element is represented by a pair of tags, typically an opening tag and a closing tag. The content of the element is placed between these tags.

HTML Tags:

The components used to design the structure of the websites are called HTML Tags. HTML tags are the building blocks of HTML elements. Tags are used to define and structure content within an HTML document. They are always enclosed in angle brackets (< >), and most tags come in pairs with an opening tag and a closing tag.

Hello World in HTML:

The "Hello, World!" program is a simple program traditionally used to introduce beginners to a new programming language or technology. In HTML, it's quite straightforward. Here's a simple "Hello, World!" example in HTML:

A code editor window with a dark blue background and a light blue border. The title bar at the top says 'helloworld.html'. The code is written in a light blue font and is as follows:

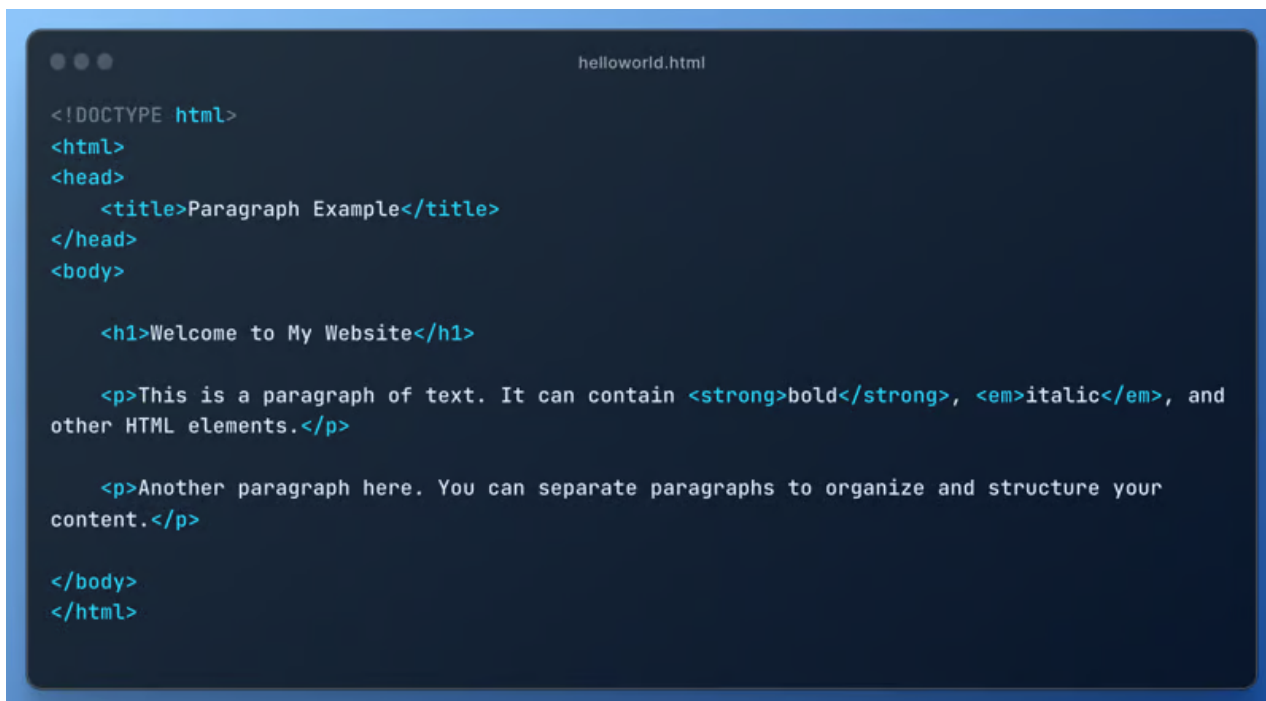
```
<!DOCTYPE html>
<html>
<head>
  <title>Hello, World!</title>
</head>
<body>

  <h1>Hello, World!</h1>

</body>
</html>
```

Paragraph Element:

The `p` element in HTML is used to define paragraphs of text. It is a block-level element, meaning it starts on a new line and typically stretches the full width of its container. Here's an example of how the `p` element is used:



```
helloworld.html

<!DOCTYPE html>
<html>
<head>
  <title>Paragraph Example</title>
</head>
<body>

  <h1>Welcome to My Website</h1>

  <p>This is a paragraph of text. It can contain <strong>bold</strong>, <em>italic</em>, and
other HTML elements.</p>

  <p>Another paragraph here. You can separate paragraphs to organize and structure your
content.</p>

</body>
</html>
```

HTML no spacing:

In HTML, consecutive spaces (including tabs and newline characters) are collapsed into a single space when rendering the content in a browser.

Nesting:

In HTML, nesting refers to placing one HTML element inside another. You can create nested structures by enclosing elements within other elements. Here's an example of nesting in HTML:

```
nestedTags.html

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Nested Tags Example</title>
</head>
<body>

  <div>
    <h1>This is a heading</h1>
    <p>This is a paragraph of text.</p>

    <ul>
      <li>Item 1</li>
      <li>Item 2</li>
    </ul>
  </div>

</body>
</html>
```

Heading Elements:

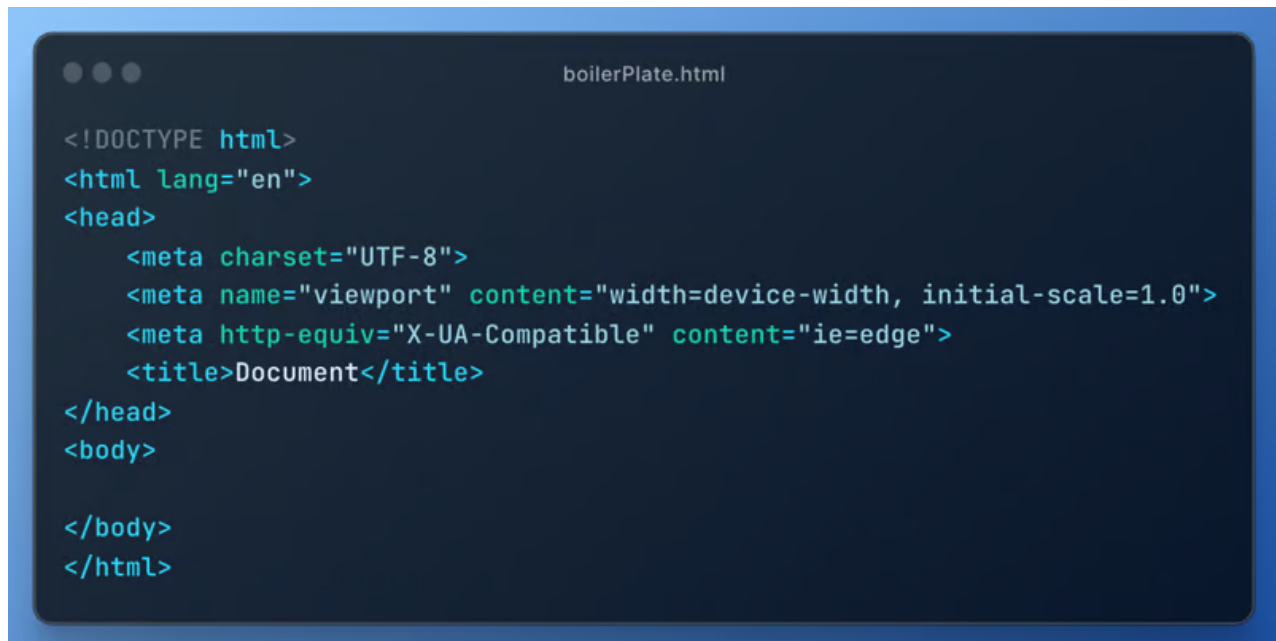
Heading elements in HTML are used to define headings or titles within a document. HTML provides six levels of heading elements, h1 to h6, where h1 represents the highest level of heading and h6 represents the lowest.

```
heading.html

<h1>This is a Heading 1</h1>
<h2>This is a Heading 2</h2>
<h3>This is a Heading 3</h3>
<h4>This is a Heading 4</h4>
<h5>This is a Heading 5</h5>
<h6>This is a Heading 6</h6>
```

HTML Boilerplate:

An HTML boilerplate is a standardized, basic HTML template that provides a starting point for building web pages. It includes essential elements and meta-information that are commonly used in web development. Here is a simple HTML5 boilerplate:



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
</head>
<body>

</body>
</html>
```

Emmet:

Emmet is a web development toolkit that greatly enhances HTML and CSS coding productivity. It provides a shorthand syntax for rapidly generating HTML and CSS code. Emmet is supported by many code editors and IDEs, making it a popular tool among web developers.

Cheatsheet: <https://docs.emmet.io/cheat-sheet/>

Indentation:

Indentation is the practice of adding spaces or tabs at the beginning of lines to visually represent the structure and hierarchy of code. In HTML, proper indentation improves code readability and makes it easier for developers to understand the document's structure. While HTML doesn't require specific indentation for functionality, it is considered a best practice for maintaining clean and organized code.

Lists in HTML:

In HTML, lists are used to group together related pieces of information. There are three main types of lists: unordered lists, ordered lists, and definition lists.

Unordered Lists (ul): An unordered list is a list of items where the order of the items doesn't matter. It is typically represented with bullet points.

html Copy code

```
<ul>
  <li>Item 1</li>
  <li>Item 2</li>
  <li>Item 3</li>
</ul>
```

The rendered output will look like:

- Item 1
- Item 2
- Item 3

Ordered Lists (ol): An ordered list is a list of items where the order of the items is significant. It is typically represented with numbers.

html Copy code

```
<ol>
  <li>First item</li>
  <li>Second item</li>
  <li>Third item</li>
</ol>
```

The rendered output will look like:

1. First item
2. Second item
3. Third item

List Items (li): The li (list item) element is used to define individual items within a list, whether it's an unordered or ordered list.

Definition Lists (dl): A definition list is used to define terms and their corresponding definitions.

```
<dl>
  <dt>Term 1</dt>
  <dd>Definition 1</dd>
  <dt>Term 2</dt>
  <dd>Definition 2</dd>
</dl>
```

The rendered output will look like:

Term 1
: Definition 1

Term 2
: Definition 2

HTML Attributes:

HTML attributes provide additional information about HTML elements and are always included in the opening tag of an element. Attributes are typically used to modify the default behavior or appearance of an element. Here are some common HTML attributes:

- **id Attribute:** Provides a unique identifier for an HTML element.
- **class Attribute:** Assigns one or more class names to an HTML element. Used for styling with CSS.
- **style Attribute:** Defines inline CSS styles for an HTML element.
- **src Attribute:** Specifies the source URL for elements like `img`, `script`, and `iframe`.
- **href Attribute:** Specifies the URL for hyperlinks (`a`).
- **alt Attribute:** Provides alternative text for images.
- **width and height Attributes:** Specifies the width and height of certain elements, such as `img`.
- **placeholder Attribute:** Defines a short hint that describes the expected value of an input field.
- **disabled Attribute:** Disables an input field or button.
- **checked Attribute:** Sets the default state of a checkbox or radio button to checked.
- **colspan and rowspan Attributes:** Specifies the number of columns or rows a table cell should span.
- **target Attribute:** Specifies where to open a link. Common values include `_blank` (open in a new tab or window) and `_self` (open in the same tab or window).

html

Copy code

```
<dl>
  <dt>Term 1</dt>
  <dd>Definition 1</dd>
  <dt>Term 2</dt>
  <dd>Definition 2</dd>
</dl>
```

The rendered output will look like:

Term 1
: Definition 1

Term 2
: Definition 2

Anchor Element:

The anchor (a) element in HTML is used to create hyperlinks, allowing users to navigate to different web pages or resources. The a element is one of the fundamental components for building a navigation system on a website.

html

Copy code

```
<a href="https://www.example.com" target="_blank">Visit Example.com</a>
```

Image Element:

The image (img) element in HTML is used to embed images into a web page. It allows you to display graphics, photos, icons, or any other visual content on your website.

html

Copy code

```

```

Br Tag:

The br tag in HTML is used to insert a line break or line break-like spacing within the content. It is a self-closing tag, meaning it doesn't have a closing tag. Here's an example of how the br tag is used:

This text has a line break.
Here is the next line of text.

Bold, Italic and Underline Tag:

- The **b** tag is used for bold text.
- The **i** tag is used for italic text.
- The **u** tag is used for underlined text.

```
tags.html

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Text Styling Example</title>
</head>
<body>

  <p>This is <b>bold</b> text.</p>
  <p>This is <i>italic</i> text.</p>
  <p>This is <u>underlined</u> text.</p>

</body>
</html>
```

Comments in HTML:

In HTML, you can use comments to add explanatory notes or annotations to your code. Comments are not displayed on the web page and are ignored by browsers. They are meant for developers or anyone reading the code to provide additional information. Here's how you can add comments in HTML:

- **Single-line comments** are enclosed within `<!--` and `-->`.
- **Multi-line comments** start with `<!--` and end with `-->`, and they can span multiple lines.

HTML is not CASE SENSITIVE:

HTML, by itself, is not case-sensitive. HTML tags and attributes are case-insensitive, meaning you can use uppercase or lowercase letters interchangeably. For example, p and P are treated the same way in HTML, and browsers will interpret them as paragraphs.

However, it's essential to note that while HTML tags and attributes are not case-sensitive, other web technologies used in conjunction with HTML, such as CSS and JavaScript, may have case-sensitive rules. For consistency and best practices, it's common to use lowercase for HTML tags and attributes.

Hr Tag:

The hr tag in HTML is used to create a horizontal rule or line, typically used to separate content or sections on a webpage. The hr tag is a self-closing tag, meaning it doesn't have a closing tag.

HTML Entities:

HTML entities are special codes used to represent characters that have a specific meaning in HTML or characters that cannot be easily represented directly. These entities are especially useful when you want to display reserved characters or characters that have a special significance in HTML, such as <, >, &, and others.

Here are some commonly used HTML entities:

- <: Represents the less-than symbol <.
- >: Represents the greater-than symbol >.
- &: Represents the ampersand &.
- ": Represents the double quotation mark ".
- ': Represents the apostrophe or single quotation mark '.

Inline vs Block:

In HTML and CSS, elements are categorized into two main display types: inline and block. These display types affect how elements are visually rendered on a webpage and how they interact with other elements.

Inline Elements: Inline elements are those that only take up as much width as necessary and do not start on a new line. They flow within the content and only take up as much width as needed. Examples of inline elements include span, a, strong, em, img, and br.

Block Elements: Block elements, on the other hand, start on a new line and stretch the full width of their container. They create a "block" of content. Examples of block elements include div, p, h1 to h6, ul, ol, and li.

Inline-Block Elements: There's also a third category called inline-block, which combines aspects of both inline and block elements. An inline-block element will flow with the content like an inline element but can have block-level properties like width and height.

Div Element:

The div element in HTML is a block-level container that is often used to group and structure content. It is a versatile element and does not have any specific semantic meaning on its own. Instead, it is commonly used as a container for other HTML elements, providing a way to apply styling and structure to groups of content.

```
<body>

  <div class="container">
    <h1>Main Content</h1>
    <p>This is the main content of the page.</p>

    <div class="box">
      <h2>Subsection 1</h2>
      <p>This is a subsection within the container.</p>
    </div>

    <div class="box">
      <h2>Subsection 2</h2>
      <p>Another subsection within the container.</p>
    </div>
  </div>

</body>
```

Span Element:

The span element in HTML is an inline container that is often used to apply styles or scripting to a specific portion of text within a larger block of content. It is a generic, non-semantic container similar to the div element but is inline by default, meaning it does not start on a new line.

```
<body>

  <p>This is a <span class="highlight">highlighted</span> word in a sentence.</p>

</body>
```

Hr Tag:

The hr tag is used to create a horizontal rule (line).

Sub & Sup Tag:

The sub and sup tags in HTML are used to define subscript and superscript text, respectively. Subscript text is typically smaller and positioned below the baseline, while superscript text is smaller and positioned above the baseline.

Semantic Markup:

Semantic markup refers to the practice of using HTML tags in a way that conveys the meaning or structure of the content, rather than just its appearance. Semantic HTML elements provide information about the structure and purpose of the content, making it more accessible, search engine-friendly, and maintainable.

Here are some examples of semantic HTML elements:

- **header**: Represents the header of a section or a page.
- **nav**: Represents a navigation menu.
- **main**: Represents the main content of the document.
- **article**: Represents an independent piece of content that can be reused and distributed.
- **section**: Represents a thematic grouping of content.
- **aside**: Represents content that is tangentially related to the content around it.
- **footer**: Represents the footer of a section or a page.

```

semanticTags.html

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Semantic Markup Example</title>
</head>
<body>

  <header>
    <h1>Website Title</h1>
  </header>

  <nav>
    <ul>
      <li><a href="#home">Home</a></li>
      <li><a href="#about">About</a></li>
      <li><a href="#contact">Contact</a></li>
    </ul>
  </nav>

  <main>
    <article>
      <h2>Article Title</h2>
      <p>This is the main content of the article.</p>
    </article>

    <section>
      <h2>Section Title</h2>
      <p>This is a thematic section of content.</p>
    </section>
  </main>

  <aside>
    <h2>Related Content</h2>
    <p>This content is related to the main content.</p>
  </aside>

  <footer>
    <p>&copy; 2024 Website Name. All rights reserved.</p>
  </footer>

</body>
</html>

```

Tables:

Tables in HTML are used to organize and display data in rows and columns. The basic structure of an HTML table involves the use of the `table`, `tr` (table row), `th` (table header), and `td` (table data) elements.

Semantic in Tables: Semantic HTML is about using HTML elements to convey the meaning or structure of the content. When working with tables, semantic markup involves using appropriate table-related elements to enhance the readability and accessibility of the table data. Here's an example of how to use semantic markup in tables:

```

tables.html

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Semantic Table Example</title>
</head>
<body>

  <h2>Employee Information</h2>

  <table>
    <caption>Employee List</caption>
    <thead>
      <tr>
        <th scope="col">Name</th>
        <th scope="col">Position</th>
        <th scope="col">Department</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>John Doe</td>
        <td>Software Engineer</td>
        <td>Development</td>
      </tr>
      <tr>
        <td>Jane Smith</td>
        <td>Marketing Specialist</td>
        <td>Marketing</td>
      </tr>
      <tr>
        <td>Bob Johnson</td>
        <td>HR Manager</td>
        <td>Human Resources</td>
      </tr>
    </tbody>
  </table>

</body>
</html>

```

Colspan: The colspan attribute in HTML is used to specify the number of columns a table cell should span (extend across) horizontally. This attribute is particularly useful when you want to merge multiple adjacent cells into a single, larger cell.

Rowspan: The rowspan attribute in HTML is used to specify the number of rows a table cell should span (extend across) vertically. This attribute is particularly useful when you want to merge multiple cells in a column into a single, larger cell.

Forms:

Forms in HTML are used to collect user input, such as text, selections, and buttons. The form element is the container for all form elements, and various form controls like text fields, checkboxes, radio buttons, and buttons are used to create interactive user interfaces.

Action Attribute: The action attribute in an HTML form element specifies the URL or destination where the form data should be sent when the user submits the form. The action attribute is a required attribute for the form element, and it typically points to a server-side script or a URL endpoint that handles the form submission.

Input: The input element in HTML is a versatile form control that allows users to enter data or make selections. The type of input is specified using the type attribute.

Type: The type attribute in the input element specifies the type of input control that should be rendered. Different values of the type attribute determine the behavior and appearance of the input element.

Placeholder: The placeholder attribute in the input element is used to provide a short hint or example of the expected format or value of the input field. It is typically used to give users guidance on what information should be entered into the field.

Label: The label element in HTML is used to associate a label with a form control, providing a more accessible and user-friendly experience. The for attribute in the label element is used to explicitly associate the label with a specific form control by referencing the id attribute of that control.

Button: In HTML, the button element is used to create a clickable button. Buttons can be used for various purposes, such as submitting forms, triggering JavaScript functions, or simply creating interactive elements on a webpage.

Name Attribute: The name attribute in HTML is used to give a name to a form control, allowing it to be referenced and processed on the server side when the form is submitted. Each form control within a form should have a unique name attribute.


```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>HTML Form Example</title>
</head>
<body>

  <h2>Contact Us</h2>

  <form action="/submit-form" method="post">
    <!-- Text Input -->
    <label for="name">Name:</label>
    <input type="text" id="name" name="name" required>

    <br>

    <!-- Email Input -->
    <label for="email">Email:</label>
    <input type="email" id="email" name="email" required>

    <br>

    <!-- Radio Buttons -->
    <label>Gender:</label>
    <input type="radio" id="male" name="gender" value="male" checked>
    <label for="male">Male</label>
    <input type="radio" id="female" name="gender" value="female">
    <label for="female">Female</label>

    <br>

    <!-- Checkbox -->
    <input type="checkbox" id="subscribe" name="subscribe" checked>
    <label for="subscribe">Subscribe to Newsletter</label>

    <br>

    <!-- Textarea -->
    <label for="message">Message:</label>
    <textarea id="message" name="message" rows="4" required></textarea>

    <br>

    <!-- Submit Button -->
    <button type="submit">Submit</button>
  </form>

</body>
```

```
</html>
```

Input Element: The input element in HTML can have various types based on the type attribute, which specifies the kind of input the element represents. Here's an overview of common input types:

- Text Input
- Password Input
- Radio Buttons
- Checkbox
- Email Input
- Number Input
- Date Input
- File Upload
- Submit Button
- Reset Button
- Hidden Input

```
input.html

<!-- Text Input -->
<input type="text" name="username" id="username" placeholder="Enter your username" required>

<!-- Password Input -->
<input type="password" name="password" id="password" placeholder="Enter your password"
required>

<!-- Radio Buttons -->
<input type="radio" name="gender" value="male" id="male">
<input type="radio" name="gender" value="female" id="female">

<!-- Checkbox -->
<input type="checkbox" name="subscribe" id="subscribe">

<!-- Email Input -->
<input type="email" name="email" id="email" placeholder="Enter your email" required>

<!-- Number Input -->
<input type="number" name="quantity" id="quantity" min="1" max="10" placeholder="Enter a
number" required>

<!-- Date Input -->
<input type="date" name="dob" id="dob" required>

<!-- File Upload -->
<input type="file" name="file" id="file" accept=".jpg, .jpeg, .png" required>

<!-- Submit Button -->
<input type="submit" value="Submit">

<!-- Reset Button -->
<input type="reset" value="Reset">

<!-- Hidden Input -->
<input type="hidden" name="secret" value="hiddenvalue">
```

MDN Docs: <https://developer.mozilla.org/en-US/docs/Web/HTML>

Top comments (0)

Subscribe



Are you sure you want to hide this comment? It will become hidden in your post, but will still be visible via the comment's [permalink](#).

For further actions, you may consider blocking this person and/or [reporting abuse](#)